UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/836,226	04/18/2001	Kevin John Moore	169.2025	6435

5514   7590   05/09/2005

FITZPATRICK CELLA HARPER & SCINTO
30 ROCKEFELLER PLAZA
NEW YORK, NY 10112

EXAMINER
WOODS, ERIC V

ART UNIT	PAPER NUMBER
2672	

DATE MAILED: 05/09/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/836,226	MOORE, KEVIN JOHN
	Examiner	Art Unit
	Eric V Woods	2672

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 February 2005*.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-33* is/are pending in the application.

　　4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-33* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

　　Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

　　Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

　　a)☒ All   b)☐ Some *   c)☐ None of:

　　　1.☒ Certified copies of the priority documents have been received.

　　　2.☐ Certified copies of the priority documents have been received in Application No. _____.

　　　3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

　　* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
　　Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
　　Paper No(s)/Mail Date _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

Response to Arguments

1.   Applicant's arguments, see Pages 4-8 of the Remarks, filed 04 February 2005, with respect to the rejection(s) of claims 1-6, 12-17, and 23-28 under 35 U.S.C. 112, second paragraph, have been fully considered and are persuasive. Therefore, the rejection has been withdrawn.

The rejections of claims 1-8 and 18-19 under 35 U.S.C. 101 have **NOT** been withdrawn.

The rejections of claims 9-17 and 20-22 under 35 U.S.C. 101 have been withdrawn.

The rejections of claims 1-33 under 35 U.S.C. 102(b) and 103(a) have been withdrawn in view of applicant's amendment.

2.   However, upon further consideration, new ground(s) of rejection are made in view of the original references.

3.   Firstly, the rejections under 35 U.S.C. 101 are discussed as below.

To begin, claims 1-8, 18-19, and 23-30 fail the relevant tests to be statutory subject matter. Firstly, they are not directed to a "concrete, tangible, and practical result", which in *State Street* the Court found was the share price. In claims 1-8, 18-19, and 23-30, no clear end result is found. On the other hand, after further consideration, claims 9-17, 20-22, and 30-33 are found to be directed to a technologically embodied method that produces a tangible, practical result – rendering an expression tree into a raster pixel image having a plurality of scan lines and pixel locations, and then

executing those instructions. Please note that the claims 1-8, 18-19, and 23-30 recite

inter alia generating instructions, but never their execution. As such, the instructions

are never used. Although the claims do recite "pixel locations" these are merely the

abstract locations of the polygons in computer memory; such items are not drawn or

rendered in any manner. The claims are broadly reciting a data structure *per se*

The claims further fail the Freeman-Walker-Abele test, and fail to perform any

statutory pre- or post-computer processes or processing or activities that would place

them within the *Abele* safe harbor. Further, they are simply not directed to a practical

application; methods of traversing abstract data structures are simply not practical

applications. There is no practical application of the algorithm (e.g. inorder traversal of a

binary tree or directed acyclic graph) to a technological art – See Alappat, 33 F.3d at

1543, 31 USPQ2d at 1556-57 (quoting Diamond v. Diehr, 450 U.S. at 192, 209 USPQ

at 10). There is also no external application, e.g. external source of input, of or to the

system that denies it protection under *Abele*, and under *Arrhythmia*, 958 F.2d at 1056,

22 USPQ2d at 1036).

Again, turning to relevant case law, e.g. AT &T, 172 F.3d at 1358, 50 USPQ2d at

1452, the method does not recite a step or act that produces something that is tangible

or useful. Applicant's arguments further only support this holding for the claims in

question. Examiner acknowledges that for certain of the claims, a practical result is

created, e.g. the resulting instructions are executed and used to render an image, and

as such the rejection under 35 U.S.C. 101 has been withdrawn **for those claims only,**

as stated above.

For the other claims, applicant states, "The method steps refer explicitly to groups of pixel locations and to generating instructions for those graphic objects active at those pixel locations" (see page 3 of Remarks). However, applicant is being somewhat inaccurate here – these "pixel locations" refer to coordinate locations in a memory that represent those pixels, but not those pixels on a display. There is no recitation of actual rendering an image or executing the instructions that have been generated. As long as those polygons (the recited "groups of pixel locations" or "graphic objects active at pixel locations") are still in memory and/or a frame buffer, and not actually directed to a display, they are still being manipulated, and as such prima facie represent data structures per se and abstract ideas being manipulated. Nowhere in the recited claims is there a practical, tangible, concrete result, not even a Boolean value as applicant claims *State Street* allows (examiner disagrees with the conclusion that a Boolean per se would satisfy *State Street*, but that is not the point here).

As such, all protections that applicant could appeal to (*Abele* safe harbor, *Alappat* tangible machine, *State Street* tangible method, *AT&T* method, etc.) have been thoroughly discussed above and are proven to be inapplicable and/or irrelevant to the present case.

4.   Secondly, applicant is reminded that while Fraser may evaluate the entire expression tree (first paragraph, Arguments, page 7), and while Politis may very well optimize the entire tree prior to rendering (page 9, Arguments), the computation of state "on-the-fly" by the present invention is irrelevant to the present discussion. Even if the present version of claim 7 only tracks changes as far as they relate to nodes prior to

generating instructions, that limitation – and the computation of state "on-the-fly" – are not in the claims. Indeed, the only amendment to claim 7 relates primary to adding 'wherein the groups are bounded by the object edges' and changing the wording to explicitly state that nodes are ignored under certain circumstances, rather than saying that nodes are traversed only under certain circumstances. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e. those stated above) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). Therefore, applicant's arguments are spurious and are thusly not persuasive in light of *In re Van Geuns*.

Claim Rejections - 35 USC § 101

5.   The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

6.   Claims 1-8, 18-19, and 23-30 are rejected under 35 U.S.C. 101 because they are directed to non-statutory subject matter (see the above paragraphs in Response to Arguments), as they are the computer programs implementing the method claims 1-7. Further, the amendment by applicant brings out the nature of the claims as being non-statutory. Thusly, this rejection is proper for a final action.

Claim Rejections - 35 USC § 103

7.   The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

8.   Claims 1-6, 12-17, and 23-28 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Fraser et al (AU 9947508) in view of Katzenberger (US 5,970,496)

and Politis (5,745,121).

Please note: claims 12-17 are treated with 1-6, as 12-17 are the apparatus

implementing 1-6; claims 23-28 are referenced to 1-6 and 12-17 with the additional

limitation of computer-readable media covered beforehand. As to claims 23-28, they

are rejected as above with the additional limitation of computer-readable media met by

claim 23 of Politis (see col. 25, lines 34-35).

The previous rejection of the claims is repeated below. Then, the additional

limitation added by amendment of "wherein the groups are bounded by the object

edges" is addressed.

9.   As to claims 1, 12, and 23, reference Politis discloses the use of expression trees

(see col. 8, lines 16-40). Furthermore, the "expression trees" shown in Figures 16-18

are valid forms of trees, which are by definition undirected acyclic graphs. Next, Politis

teaches that these trees are decomposed into lower-level instructions (see col. 4, lines

3-8). Politis teaches these graphs as trees, but (see Figs. 7, 8, 16, and others) they

have leaf nodes (see col. 11, lines 30-33) and parents (see col. 14, lines 25-32). The

parent nodes are found to be operators (element 26, Fig. 17) as shown in Figs. 16, 17,

and 18 (see col. 13, lines 33-60). The leaf or descendent nodes are shown in the above

figures (elements 20, 24, etc.) and are representing graphical objects such as letters

(see col. 13, lines 33-60). Therefore, all the recited portions of the preamble to claim 1

and 12 are met. Rejection of the method steps follows below.

Reference Politis teaches that there are separate groups of pixel locations

determined (see col. 4, lines 15-32), and such selections are illustrated in Figs. 30 – 32,

as regions of pixels are shown with their bounding boxes (see col. 11, lines 3-4).

Furthermore, the operators are all described as operating within a region limited by such

a box (see col. 11, lines 3-4). The claim recites, "...wherein the said portion of the

directed adjacency graph is that portion which passes data up the directed adjacency

graph..." (Ellipses added). This language is with used with respect to the determining of

the graph to be operated upon, which is clearly shown in Fig. 17 as graphical regions

are selected and would *prima facie* pass information up the tree or graph containing

them. Also, Politis very clearly teaches the determining of active pixel areas (col. 9,

lines 31-42) and the updating of it – that is, inactive or completed instructions are

pruned from the list. Only the active leaf nodes would be placed into the active list for a

given location during the drawing process - that is, as the rastering shifted, the active

instruction pool would update. Again, the above means that the tree is traversed as the

rasterizing / scanning process occurs; the instructions are generated and put into the

active list (see col. 9, 12-25 and 31-52). Lastly, the generation step for the trees is

described in the lines referenced above, e.g. the trees are constructed during each line

scan.

Reference Katzenberger discloses the use of directed acyclic graphs to store

graphical information in Figs. 2 and 4B, particularly Fig. 4B (see col. 1, lines 9-11).

Parent and child nodes are used in this invention (see col. 2, lines 53-65); child nodes

are the same thing as the "leaf" nodes referenced by applicant (see col. 1, lines 55-65).

Furthermore, Katzenberger teaches the use of operators for performing operations on

graphical areas – Fig. 14 shows the use of various operators such as 'cno' and 'ctg' that

meet the requirement of the claim, even if they primarily emphasize the data

relationships. As seen in Fig. 4B, Katzenberger shows the simplest graphical elements

as the leaf nodes. Therefore, it would be obvious to one of ordinary skill in the art to

combine the instruction trees of Politis with the graph structures of Katzenberger. The

motivation would be to allow compiler-style optimization of rasterizing instructions and to

enable the effective use of directed acyclic graphs as discussed by Politis (see Politis

col. 10, lines 1-11 and 37-40) and also to derive the benefits of the DAG storage

structures as shown by Katzenberger in those figures referenced above.

The additional limitation of having the groups bounded by the object edges is met

by the addition of the Fraser reference. Fraser in Figure 8 teaches the rendering of at

least one graphic object (80, 90) described by at least one edge (82-86, 92-98) into a

raster pixel image (78) having a plurality of scan lines and a plurality of pixel locations

on each scan line. For each scan line, coordinates of intersection of those edges of the

objects that intersect the scan line are determined in a predetermined order (taken from

Fraser abstract as a summary).

The motivation for combination with Katzenberger and Politis is as follows:

Politis teaches the use of bounding boxes to draw polygons, which is less efficient than

the method of Fraser. Bounding boxes suffer from the limitations of redrawing

background pixels in the bounding box area unnecessarily and have other problems, as

set forth in Fraser specification pages 1-2, as they typically use something similar to the

"painter's algorithm". However, Fraser's method is faster and more efficient, and only

draws the areas necessary (pages 5-7 of the specification). Therefore, it would be

obvious to modify the combination of Politis and Katzenberger in view of the teachings

of Fraser to use edge-delineated polygons or groups of pixels versus the previously

used bounded boxes.

For the dependent claims below, the motivation for combining Fraser is taken

from the parent claim and incorporated by reference.

10.   As to claims 2, 13, and 24, references Politis and Katzenberger teach all the

limitations. Reference Katzenberger teaches a table for storing graphical instructions

for acting on image components in Fig. 9, thus meeting the details recited in claim 2.

See claim 1 for motivation and combination.

11.   As to claims 3, 14, and 25, references Politis and Katzenberger teach all the

limitations. Reference Politis teaches the use of a "Cliplist" in Fig. 24 that contains a

listing of elements to be moved. This list is stored in a one-column tabular format as

shown in the Figure, and an operator does perform this action (see col. 3, lines 20-40).

This data is constructed as the tree is traversed and could easily be inserted into the

table of Katzenberger shown in Fig. 9. See claim 1 for motivation and combination.

12.   As to claims 4, 15, and 26, reference Katzenberger discusses traversing a

directed acyclic graph (see col. 18, lines 8-15). This step is executed as part of an

operation on the data structure. As discussed in claim 1, the active nodes are read in

for each pass and updated, therefore both the operators and the active leaf nodes are

traversed and translated into instructions, thus meeting the requirements of the claim.

See claim 1 for motivation and combination.

13.   As to claims 5, 16, and 27, reference Politis discloses the use of expression trees

(see col. 8, lines 16-40). Furthermore, the "expression trees" shown in Figures 16-18

are valid forms of trees, which are by definition undirected acyclic graphs. Next, Politis

teaches that these trees are decomposed into lower-level instructions (see col. 4, lines

3-8). Politis teaches these graphs as trees, but (see Figs. 7, 8, 16, and others) they

have leaf nodes (see col. 11, lines 30-33) and parents (see col. 14, lines 25-32).

Examiner interprets these terms, e.g. leaf and parent nodes, to have the normal

meanings associated with them in graph theory. The parent nodes are found to be

operators (element 26, Fig. 17) as shown in Figs. 16, 17, and 18 (see col. 13, lines 33-

60). The leaf or descendent nodes are shown in the above figures (elements 20, 24,

etc.) and are representing graphical objects such as letters (see col. 13, lines 33-60).

Therefore, it would be obvious to one of ordinary skill in the art to combine the trees of

Politis with the graphic forms of Katzenberger. The motivation would be to have a tree

structure to optimize as per claim 1 (see claim 1 references and col. 4, lines 1-10).

14.   As to claims 6, 17, and 28, reference Politis teaches the use of binary operators

on the expression trees – see Figs. 28-29, for example. See claim 1 for combination

and motivation.

15.   Claims 7–11, 18-22, and 29-33 are rejected under 35 U.S.C. 103(a) as

unpatentable over Fraser in view of Politis.

16.   As to claims 7, 18, and 29, reference Fraser teaches most of the limitations as

set forth below.   Expression trees are discussed and all the requirements of the

preamble met (see pg. 45, lines 10-26; pg. 46, lines 1-5). Fraser teaches the selection

of a region of pixels and their use as a leaf node-primitive and the use of an operator as

a parent node (see pg. 45, lines 10-26; pg. 46, lines 1-5). Further, all the cases of

active and inactive nodes are covered (see pg. 45, lines 23-26). The process of

traversing the nodes, setting active flags, and similar is all covered in pages 47-50 of

the Fraser specification. Furthermore, Fraser teaches the generating of operator

instructions (see pg. 30, lines 14-2).

Politis discloses the use of expression trees (see col. 8, lines 16-40).

Furthermore, the "expression trees" shown in Figures 16-18 are binary; that is, as

evinced in Figs. 17 and 18 particularly, each parent node has both a left and a right

descendent node, which each consist of a graphical object. The parent node is an

operator, representing a common, overlapping region, e.g. nodes 37, 36, and 26 on Fig.

17, and the descendent nodes represent graphical objects, e.g. nodes 28-30 and others

on Fig. 17. Next, Politis teaches that these trees are decomposed into lower-level

instructions (see col. 4, lines 3-8). Politis teaches these graphs as trees, but (see Figs.

7, 8, 16, and others) they have leaf nodes (see col. 11, lines 30-33) and parents (see

col. 14, lines 25-32). Examiner interprets these terms, e.g. leaf and parent nodes, to

have the normal meanings associated with them in graph theory. The parent nodes are

found to be operators (element 26, Fig. 17) as shown in Figs. 16, 17, and 18 (see col.

13, lines 33-60). The leaf or descendent nodes are shown in the above figures

(elements 20, 24, etc.) and are representing graphical objects such as letters (see col. 13, lines 33-60). Therefore, all the recited portions of the preamble to claim 7 are met. Rejection of the method steps follows below.

Reference Politis teaches that there are separate groups of pixel locations determined (see col. 4, lines 15-32), and such selections are illustrated in Figs. 30 – 32, as regions of pixels are shown with their bounding boxes (see col. 11, lines 3-4). Furthermore, the operators are all described as operating within a region limited by such a box (see col. 11, lines 3-4). The selection of active or inactive regions is clearly shown in Fig. 17 as graphical regions are selected and would *prima facie* pass information up the tree or graph containing them. Also, Politis very clearly teaches the determining of active pixel areas (col. 9, lines 31-42) and the updating of it – that is, inactive or completed instructions are pruned from the list. Only the active leaf nodes would be placed into the active list for a given location during the drawing process - that is, as the rasterizing shifted, the active instruction pool would update. Again, the above means that the tree is traversed as the rasterizing / scanning process occurs; the instructions are generated and put into the active list (see col. 9, 12-25 and 31-52). Furthermore, the tree is traversed recursively, thusly all the nodes are determined active or not – since the traversal is done immediately before execution to save memory, this means that it is constantly traversed to determine active and inactive nodes for compile-time and real-time optimization (see col. 8, lines 15-40).

The order of traversal of the nodes, which is explicitly recited in claim 7, corresponds to the preorder traversal taught by Politis (see col. 14, lines 25-55). The

generation step for the trees is described in the lines referenced above, e.g. the trees

are constructed during each line scan. For further exploration of the issues not explicitly

discussed here, please see the below rejection under U.S.C. 103 in the Politis section.

The additional limitation of having the groups bounded by the object edges is met

by the addition of the Fraser reference. Fraser in Figure 8 teaches the rendering of at

least one graphic object (80, 90) described by at least one edge (82-86, 92-98) into a

raster pixel image (78) having a plurality of scan lines and a plurality of pixel locations

on each scan line. For each scan line, coordinates of intersection of those edges of the

objects that intersect the scan line are determined in a predetermined order (taken from

Fraser abstract as a summary).

The limitation of ignoring nodes unless they meet certain conditions is met as

above, namely that the above stated, previously presented arguments about the order

of traversal of nodes still stands, in that the system would ignore inactive nodes and

only act upon active nodes to generate the instructions as set forth above.

The motivation for combination with Politis is as follows: Politis teaches the use

of bounding boxes to draw polygons, which is less efficient than the method of Fraser.

Bounding boxes suffer from the limitations of redrawing background pixels in the

bounding box area unnecessarily and have other problems, as set forth in Fraser

specification pages 1-2, as they typically use something similar to the "painter's

algorithm". However, Fraser's method is faster and more efficient, and only draws the

areas necessary (pages 5-7 of the specification). Therefore, it would be obvious to

modify the combination of Politis and Katzenberger in view of the teachings of Fraser to

use edge-delineated polygons or groups of pixels versus the previously used bounded boxes.

Motivation and combination for the dependent claims that follow is taken from the parent claim and incorporated by reference.

17.   As to claims 8, 19, and 30, which focus only on the details of traversing the expression tree of claim 7, the refutation of the expression tree traversal paths based on Politis as above shows that Politis teaches all these limitations, including the additional limitation of computer-readable media met by claim 23 of Politis (see col. 25, lines 34-35). Reference Fraser teaches all the limitations of this acclaim as discussed in the above rejection to claim 7 anyway.

18.   As to claims 9, 20, and 31, reference Fraser teaches most of the limitations. Fraser teaches a method of processing graphical objects to form a raster pixel image on both one and a subsequent scan line, meeting the recited requirement of a plurality of pixel locations and a plurality of scan lines (see col. 4, lines 8-26). The use of expression trees and their details are discussed in the rejections for claims 7 and 8 above. Again, such a tree is shown in Fig. 19 with binary nodes, a plurality of leaf nodes, and the other required elements. As to the method claims, see below. Also, the rejection to claim 7 is herein incorporated by reference in its entirety.

In Figs. 20A-20C, there is clearly illustrated a table that represents an expression tree. This table is clearly defined in terms of union regions, e.g. the areas of each pixel that overlap and that has active fields. Further, each row represents an entry in the table representing a binary or leaf node. The table clearly shows that each record has

both first and second fields that show whether a left or a right node is required for an

operation. This is evident in both the data_in_S∩D fields and the multiple _Active fields.

Next, the third and fourth fields that indicate whether the left or right nodes are active

are a combination of Src_Active, and one of data_in_S∩D fields.

Fraser teaches the determining of pixel locations (see col. 3, lines 23-40). The

traversing and generating steps are discussed in the rejections for claims 7 and 8 under

Fraser. Finally, Fraser teaches the execution of the instruction to render the image from

the image-rendering apparatus 20 (see Fig. 3, the output 898 from the pixel output

module 800 that does the rendering, and the entire rendering apparatus 20).

The additional limitation of having the groups bounded by the object edges is met

by the addition of the Fraser reference. Fraser in Figure 8 teaches the rendering of at

least one graphic object (80, 90) described by at least one edge (82-86, 92-98) into a

raster pixel image (78) having a plurality of scan lines and a plurality of pixel locations

on each scan line. For each scan line, coordinates of intersection of those edges of the

objects that intersect the scan line are determined in a predetermined order (taken from

Fraser abstract as a summary).

The limitation of ignoring nodes unless they meet certain conditions is met as

above, namely that the above stated, previously presented arguments about the order

of traversal of nodes still stands, in that the system would ignore inactive nodes and

only act upon active nodes to generate the instructions as set forth above.

The motivation for combination with Politis is as follows: Politis teaches the use

of bounding boxes to draw polygons, which is less efficient than the method of Fraser.

Bounding boxes suffer from the limitations of redrawing background pixels in the

bounding box area unnecessarily and have other problems, as set forth in Fraser

specification pages 1-2, as they typically use something similar to the "painter's

algorithm". However, Fraser's method is faster and more efficient, and only draws the

areas necessary (pages 5-7 of the specification). Therefore, it would be obvious to

modify the combination of Politis and Katzenberger in view of the teachings of Fraser to

use edge-delineated polygons or groups of pixels versus the previously used bounded

boxes.

Motivation and combination for the dependent claims that follow is taken from the

parent claim and incorporated by reference.

19.   As to claims 10, 21, and 32, reference Fraser teaches multiple fields in the table

representing the binary elements of the expression tree as discussed in the above

rejection. However, there are other fields besides the four discussed above that are

disclosed in Figs. 20A-20C. While the fields are not explicitly directed to holding the

information on overlap in the common area, the nature of the fields should be taken into

account here. The phrasing "data_in_S∩D fields" that is utilized by Fraser in his

description of the invention is such that it accommodates overlap because it represents

set relationships. If there were overlap between source and destination, it would be

indicated in these fields. Therefore, it would be valid to hold that these and the other

fields inherently hold this information and thus would meet the requirements recited by

the claim. All of the traversal requirements and subsequent portions of the claim are

discussed above for claim 9.

20.   As to claims 11, 22, and 33, reference Fraser teaches that these tables are used

to implement clipping operations, particularly the tables shown in Figs. 27B and 29A,

including the CLIP_IN and CLIP_OUT operations, specifically a clip-type flag (see col.

22, lines 20-45, regarding omitted flags that would be transferred to the level and/or

activity tables). That is, the level and activity tables discussed in Fraser could be

combined as shown in Figs. 20G-20I in discussing alternative implementations. Thusly,

the flags discussed as missing from Fig. 15 would be present and allow the fulfilling of

the recited requirements of the claim (see col. 22, lines 20-45).

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in

this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37

CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later

than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Eric V Woods whose telephone number is 571-272-

7775. The examiner can normally be reached on M-F 7:30-4:30 alternate Fridays off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Michael Razavi can be reached on 571-272-7664. The fax phone number

for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).

Eric Woods

JEFFERY BRIEN
PRIMARY EXAMINER

April 18, 2005